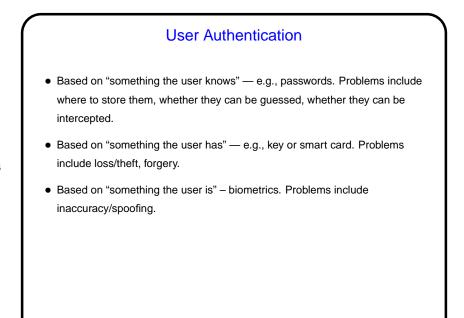
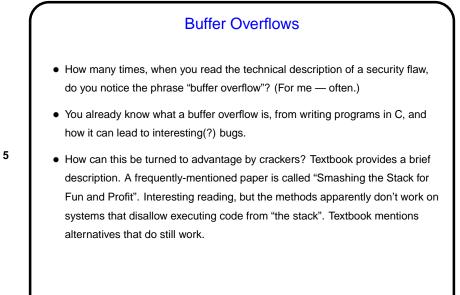
Administrivia

- As mentioned by e-mail: One more short homework on the Web. Official due date is Monday, but okay to turn in without penalty anytime Tuesday. There will be a "not accepted past" deadline for regular homeworks. There will also be a set of optional extra-credit problems, available next week. We should discuss all of this Monday?
- (Review minute essays of 11/19 and 12/01.)

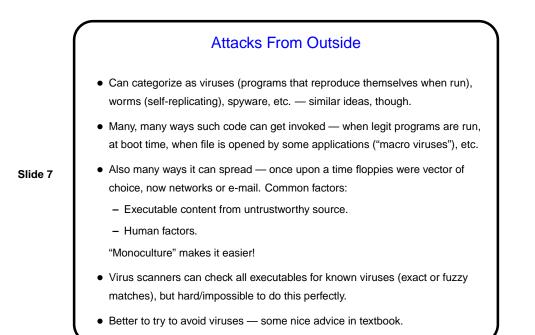
Security - Overview Goals: - Data confidentiality - prevent exposure of data. - Data integrity - prevent tampering. - System availability - prevent DOS (denial of service). Slide 2 • What can go wrong: - Deliberate intrusion - from casual snooping to "serious" intrusion. - Accidental data loss — "acts of God", hardware or software error, human error.

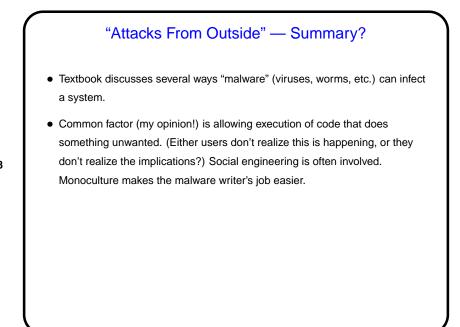


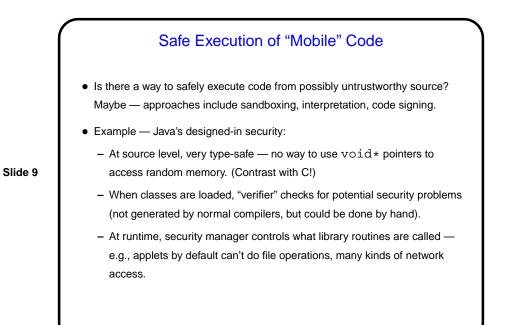
Attacks From Within Trojan horses (and how this relates to \$PATH). Login spoofing. Logic bombs and trap doors. Buffer overflows (and how this relates to, e.g, gets). Code injection attacks. And many more ...



"Attacks From Within" — Summary? Textbook discusses several ways programs can be made to do things their authors would not want and probably did not intend — buffer overflows, code injection attacks, etc. Common factor (my opinion!) is what one might call insufficient paranoia on the part of the programmers.







Fusted Systems Is it possible to write a secure O/S? Yes (says Tanenbaum). Why isn't that done? People want to run existing code. People prefer (or are presumed to prefer) more features to more security.

